REMARKS/ARGUMENTS

This Amendment is in response to the Office Action dated June 29, 2005. Claims 1-14 are pending. Claims 1-14 are rejected. Claims 1, 5, 7, and 13 have been amended. Accordingly, claims 1-14 remain pending in the present application. A request for extension of time to extend the period for reply for one month from September 29, 2005 to October 29, 2005, is requested herewith.

Claims 1 and 7 have been amended to recite that "at least one of the second private and public keys is digitally signed by the first private key of the software publisher". Claims 5 and 13 have been amended to correct a typographical error.

In the Office Action, the Examiner rejected claims 1-14 under 35 U.S.C. §102(e) as being unpatentable over U.S. Patent No. 6,898,706 (Venkatesan). Applicants respectfully disagree. Anticipation requires that a prior art reference disclose each and every claim element of the claimed invention. It is respectfully submitted that Venkatesan fails to disclose each in every claim element of the independent claims.

The present invention provides a method and system for delivery of a licensed toolset to a software publisher for creating license-managed software products. The method comprises providing an authorization process, and implementing the authorization process for both a toolset publisher and related toolset and a software publisher and related software product, whereby the same authorization process is used to obtain respective licenses. The authorization process includes creating a first public and private key pair for the software publisher, and creating a second public and private key pair for the software product, wherein at least one of the second private and public keys is digitally signed by the first private key of the software publisher. An authorization program is also created for the software program that has embedded copies of the first and second public keys. The software program and the authorization program are combined, such that when the authorization program is invoked, the authorization program obtains a license for controlling the use of the software program. The license is obtained by creating a license request, encrypting the license request using the second private key, transmitting the license request to a key authority, receiving a license from the key authority with license terms, decrypting the license, and using the license terms to control the use of the software program.

Referring now to the independent claims, in claim 1, Applicants refer broadly to private and public key pairs, while in claim 7 Applicants refer to certificates, which usually include other information besides keys. According to the preferred embodiment, the software program keys are connected to the software publisher keys so that only that publisher can allow the software program to be authorized. To accomplish this, at least one of the keys of the software program is digitally signed by the private key of the publisher. Using the public key of the publisher, the authorization program can verify that the publisher who signed its product public key is the same publisher who signed the license in response to license request.

In independent claim 7, Applicants refer to signed certificates, which have associated private keys. The public keys are part of the certificate. So in this case, the product certificate is digitally signed by the publisher private key. This allows the authorization program to verify that the publisher who signed its product certificate is the same publisher who signed the license.

In a further embodiment, the publisher of the software program can use a toolset to convert the software program into "a license-managed software product." The publisher uses the toolset and the publisher certificate to create protected software products and to create product certificates for licensing. Thus, the present invention provides a chain of certificates to authorize use of a software program through a license. To run, a software product has to verify the product certificates. Verification means verifying the certificate chain, meaning that the product certificate is cryptographically tied to the proper publisher certificate, which in turn, may be cryptographically tied to a certificate authority certificate. The elegance of the solution is that it allows the certificate authority to control how publishers use the toolset, allows publishers to control how their end-users use their protected software products, and prevents one publisher from authorizing a product from another publisher.

Venkatesan fails to teach or suggest such a software licensing mechanism, and instead teaches a content protection scheme that protects non-executable data. In essence, a relatively large number, n, of identical watermarks is embedded throughout a single software object, through use of n different secret watermark keys. Each of these watermark keys defines a starting location (e.g., in time, space or frequency) in a protected object (or, in a general sense, a pointer to a location in that object) at which a corresponding watermark appears. Once a user has downloaded the protected object through a client computer, the user then transacts with publisher's web server to obtain an electronic license, cryptographically signed by the publisher to an "enforcer" located in that computer, which specifies access rights, which the publisher accords to this client computer, and the watermark value. The client computer contains an enforcer equipped with only one of the n watermark keys. Whenever the client computer attempts to access a file containing a protected object, the enforcer examines the object using its secret watermark key. If the object contains a watermark appearing at a location specified by the enforcer's watermark key, a client operating system accesses a license database to determine whether a signed license made to the enforcer and linked, via the publisher's cryptographic signature, to this protected object resides in that database. A value of a parameter in the license must match a value of the same parameter contained in a watermark detected in the object. In that regard, the license must be signed by the publisher specified in the watermark and made to a product identification (PID) value that appears in the watermark. Thus, the watermark effectively becomes "glue" between the protected object and its license. If no such license exists, the enforcer inhibits any further access to the object. Otherwise, the enforcer determines whether the watermark value contained in the license matches that detected in the object, and, if so, permits access to the object in accordance with the rights specified in the license. The object can be either an active (executable) or a passive (content) software object. (col. 5, lines 21-57).

Unlike the present invention, in Venkatesan, there is no chaining of certificates. Thus, Venkatesan fails to teach or suggest independent claims 1 and 7. For example, Venkatesan fails to teach or suggest “creating a first public and private key pair for a software publisher”, as recited in claim 1. Venkatesan teaches that a publisher sets the value of a watermark, and that a watermarking authority (WA) embeds the watermark n times into the object in a starting location determined by corresponding different one of secret keys in order to yield the watermark object. In the present invention, watermarks are not embedded into the protected software product. And in Venkatesan, no public and private key is associated with the publisher.

Similarly, Venkatesan fails to teach or suggest “creating a second public and private key pair for a software program,... and embedding a copy of the first and second public keys in the authorization program", as recited in claim 1. Instead, Venkatesan teaches the use of secret watermark keys that define a starting location in a protected object at which a corresponding watermark appears. Although the publisher may have private/public keys, it is believed that the secret watermark keys are not analogous to the product public and private key pair because the secret watermark keys are not "paired" and different pairs are not associated with different objects. Instead, Venkatesan teaches that "all n watermark keys ... generated by the WA and are identical across all objects that are to be protected, regardless of their corresponding publishers. These keys are generated once and will be universally used for a relatively long, but finite period, for all objects, from whatever publisher or source, that are to be protected." (Col. 6, lines 2-7).

Because Venkatesan fails to teach associating a product public key and private key with a software product, Venkatesan cannot teach "digitally signing" "at least one of the product private and public keys by the first private key of the publisher", as recited in claim 1.

With respect to generating a license request, Venkatesan may teach that after a user has downloaded a watermarked object, the user, through his(her) client PC, electronically downloads from the publisher's web server an electronic license cryptographically signed by the publisher. However, it is not believed that Venkatesan teaches that "the authorization program" creates the "license request" or that the authorization program "encrypt[s] the license request using the second public key," as recited in claim 1, or "signs a formatted license request using the second public key," as recited in claim 7.

Moreover, a keyword search reveals that Venkatesan fails to teach or suggest the use of "a software toolset," and therefore fails to teach or suggest the combination of the steps (b) and (c).

Therefore, it is respectfully submitted that independent claims 1 and 7 are allowable over Venkatesan for at least these reasons.

The arguments above apply with full force and effect to the remaining dependent claims because they are based on allowable independent claims. Therefore, the dependent claims are allowable for at least the same reasons as the independent claims.

In view of the foregoing, it is submitted that claims 1-14 are allowable over the cited references. Because the secondary references stand or fall with the primary references, claims are allowable because they are dependent upon the allowable independent claims. Accordingly, Applicant respectfully requests reconsideration and passage to issue of claims 1-14 as now presented.

Applicants' attorney believes this application in condition for allowance. Should any unresolved issues remain, Examiner is invited to call Applicants' attorney at the telephone number indicated below.

Respectfully submitted,
SAWYER LAW GROUP LLP

October 31, 2005
Date

Stephen G. Sullivan
Attorney for Applicant(s)
Reg. No. 38,329
(650) 493-4540